Data Visualization and Text Mining/NLP Combined Assessment

Team 5: Diana Aycachi, Alessandro Casella, Kevin Farjallah,
Diego Polar, Vivian Soo, and Madhuri Thackeray

Dec 13th, 2021

Index

Introduction and Executive Summary

In this report, we used R programming to perform text analysis on Airbnb's description of postings across countries. In addition, we used Tableau to better visualize the relationships between countries and some of their respective numerical variables such as price, review scores, and more. Where the analysis revealed that American and European hosts are friendlier to guests and take a higher consideration of their listing being close to public transportation as well having larger sized beds. Other point of note was that the most highly rated countries also had the lower average priced listing when compared to other countries.

Tokenization and Frequencies

Assuming all prices are listed in the same currencies, the average price per night across all countries is $279. When comparing the top 10 most frequent word tokens between descriptions of below average and above average price per night, we noticed that for the higher price range, there is an added need for top entertainment, which are "shopping" and "pool" (Appendix 1). In Tableau, we can see that Hong Kong has the highest average price of $774 per night, followed by Brazil and China. Countries such as the United States, Canada, and other western countries, however, have an average price of below $200 per night (Appendix 13).

N-gram Analysis

Among the countries with below average price per night, the United States, Canada, Australia, and Portugal also have the highest average review scores compared to the other countries (Appendix 16). When we further examine the bigrams in R, we noticed that listings from these highly rated countries tend to have descriptions emphasizing on availability of nearby transportations and larger sized beds, such as "walking distance", "metro station", "queen bed", and "size bed" (Appendix 4 - 12).

Correlograms Apartments for US, Spain, Canada

Next, by creating two correlograms (Appendix 3) we compare the terms that are common and different between the United States and Canada, and between the United States and Spain.

Between the United States and Canada, we've found that "bathtub", "bbq", "bikes", "blankets", "advice", "fast" are some of the common terms they share. This suggests that the hosts of both countries offer apartments with bathtubs, space for barbeque, and possibly biking related amenities as well. In terms of differences, we can deduce that hosts from the United States might offer coffee while hosts from Canada might offer cuisine services or their properties might be located next to a variety of cuisine styles.

Between the United States and Spain, the common terms include "authentic", "bank", "cold", "club", which would mean that the hosts of these countries offer some authentic service or architecture. For terms that appear more frequently in the US, we have "closet", "coffee", "clean", "cable", "street", "bars", "subway" and more. For terms that appear more frequently in Spain we see terms like metro, balcony, barrio. However, metro can be translated as a kind of transportation like the subway, so this is under the category of common words. Hosts in Spain might offer properties located near to the plazas, which are parts of the cities that represent the downtown. On the other hand, hosts in the US put an emphasis on delivering coffee as an extra product or service, queen beds, and cleanliness.

Sentiment Analysis for Australia, Canada, and United States

For the sentiment analysis we looked at Australia, Canada, and the United States. Each country was further filtered based on high (>= $279) and low (<= $129) price ranges (Appendix 2). Their Affin, Bing and NRC scores were calculated to gain insight on how would describe a more expensive or cheaper accommodation and what facilities would entice a possible guest to make the final decision. All three countries had a higher positive score than negative for all the sentiments, giving each of them an overall positive score. Most AirBnbs in Australia were either located in Sydney or on the outskirts for easy access. Prices for accommodation near the beach or walking distance from Sydney CBD were higher. Similarly, AirBnbs offered in Canada were in/near Montreal or the Montreal Airport with a great view of the city. Lastly for the United States, the highest number of listings were in Hawaii, making beach access an important feature for the final decision. Hosts who stayed in cities like Brooklyn and New York emphasized on comfort and community, with a bonus of cultural immersion.

Dashboard analysis

Looking at the Appendix 19 and 20, for the former we can notice that countries listed from the Western side have a lower price per night on average while maintaining the highest number of properties listed, especially the United States, who has by far the largest number of properties listed with a focus on apartments. While the latter shows that Brazil who has average reviews scores, focuses on bringing more space available to the guest inside the property, especially in the form of larger amounts of bedrooms and bathrooms compared to any other country. Combining the dashboard on Appendix 20 with the bigram analysis previously done, it was discovered that higher reviewed countries, had a major focus on larger bed sizes as well as walking distance, while lower reviewed countries, focused on their on nearby attractions and amenities. A final point of note when looking at both dashboards was the fact that the most popular countries were also the ones who had a lower price compared to the others (assuming that the currencies were all standardized in dollar amounts), such as the United States or Portugal that coincidently also have a large amount of superhosts (Appendix 18).

Appendix

Appendix 1 Tokens and frequencies for the dataset

```
> low
        word   n
1   building 686
2      quiet 678
3      fully 659
4     center 582
5       full 567
6  beautiful 566
7      enjoy 564
8   spacious 431
9    perfect 405
10     clean 379
```

```
> high
        word   n
1   building 317
2      enjoy 271
3   shopping 271
4      fully 256
5      quiet 240
6   spacious 232
7       full 227
8       pool 225
9  beautiful 214
10   perfect 205
```

Appendix 2 Sentiment analysis filtered by price US-Canada_Australia

```
> bind_rows(United_States_afinn_high, United_States_bing_and_nrc_high)
  sentiment       method negative positive
1       305        AFINN       NA       NA
2       200 Bing et al.       60      260
3       289          NRC       93      382
```

```
> bind_rows(United_States_afinn_low, United_States_bing_and_nrc_low )
  sentiment       method negative positive
1       346        AFINN       NA       NA
2       214 Bing et al.      144      358
3       343          NRC      179      522
```

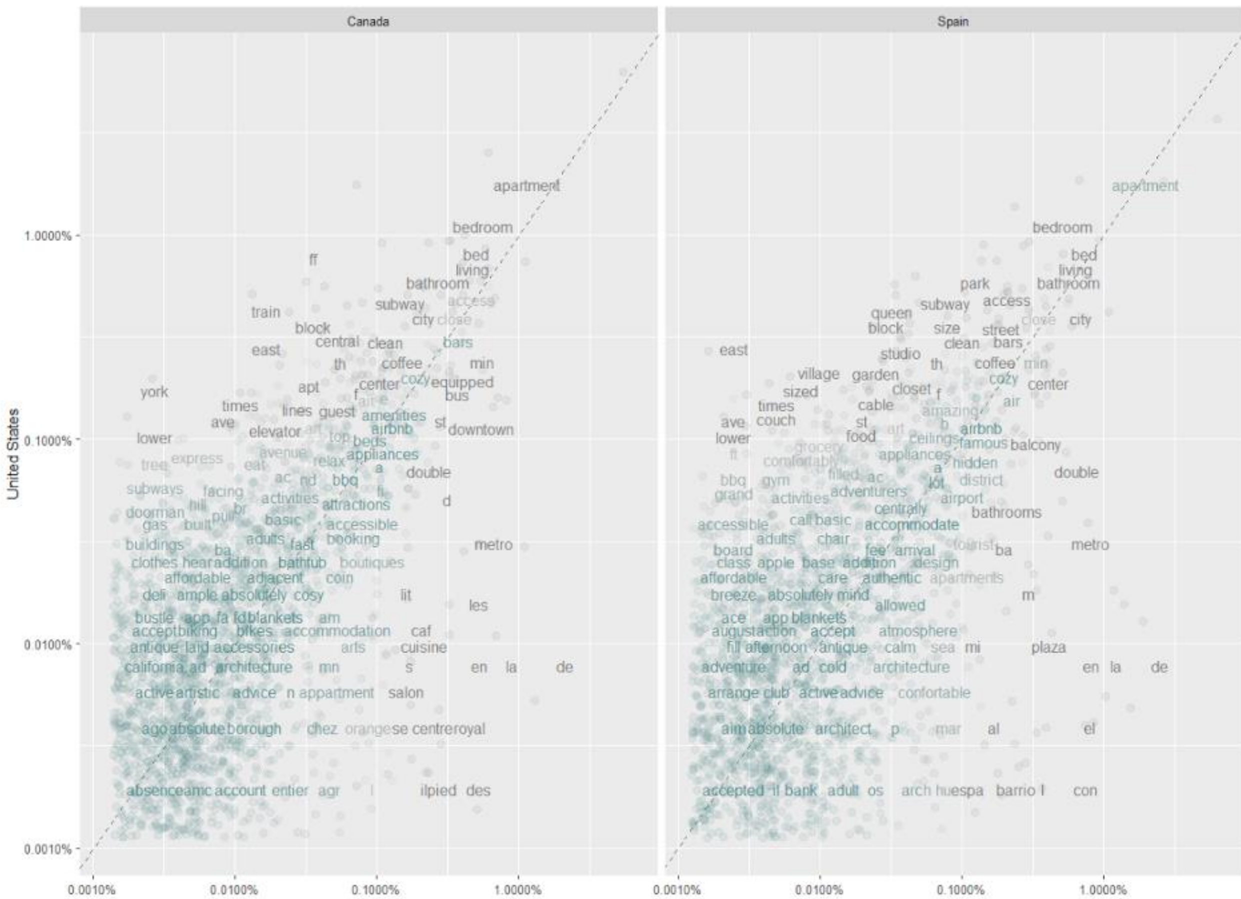
```
> bind_rows(afinn_Australia_high, bing_and_nrc_Australia_high)
  sentiment       method negative positive
1       198        AFINN       NA       NA
2       145 Bing et al.       41      186
3       164          NRC       70      234
```

```
> bind_rows(afinn_Australia_low, bing_and_nrc_Australia_low)
  sentiment       method negative positive
1       265        AFINN       NA       NA
2       170 Bing et al.       75      245
3       239          NRC      104      343
```

```
> bind_rows(afinn_Canada_high, bing_and_nrc_Canada_high)
  sentiment       method negative positive
1        67        AFINN       NA       NA
2        54 Bing et al.       11       65
3        69          NRC       11       80
```

```
> bind_rows(afinn_Canada_low, bing_and_nrc_Canada_low)
  sentiment       method negative positive
1       261        AFINN       NA       NA
2       193 Bing et al.       74      267
3       266          NRC      109      375
```

Appendix 3 - Correlogram for U.S., Canada, and Spain



Appendix 4 Bigram counts for the US

```
> bigram_counts_US
      word1        word2       Country   n
1   walking     distance United States 220
2      size          bed United States 207
3     queen         size United States 152
4    minute         walk United States 129
5   bedroom    apartment United States 119
6    washer        dryer United States 118
7   central         park United States 107
8     queen          bed United States 102
9         2      bedroom United States  96
10     flat       screen United States  94
11        1      bedroom United States  91
12   master      bedroom United States  91
13    ocean        views United States  89
14    newly    renovated United States  88
15    ocean         view United States  83
16    queen        sized United States  83
17    sized          bed United States  83
18      air conditioning United States  82
19     easy       access United States  80
20      min         walk United States  76
21     sofa          bed United States  75
22 equipped      kitchen United States  74
23   screen           tv United States  69
24    north        shore United States  68
25       sq           ft United States  68
```

Appendix 5 Bigram counts for Canada

```
> bigram_counts_Canada
       word1     word2 Country   n
1       mont     royal  Canada 179
2      montr        al  Canada 133
3         de        la  Canada 119
4    walking  distance  Canada 109
5      metro   station  Canada 104
6      salle        de  Canada  86
7       size       bed  Canada  81
8         de      bain  Canada  78
9     centre     ville  Canada  75
10      jean     talon  Canada  75
11         5   minutes  Canada  74
12  downtown  montreal  Canada  67
13       min      walk  Canada  64
14        de     montr  Canada  62
15  equipped   kitchen  Canada  62
16     queen      size  Canada  60
17   plateau      mont  Canada  57
18   minutes      walk  Canada  55
19        wi        fi  Canada  54
20        la       rue  Canada  51
21  logement       est  Canada  49
22        10   minutes  Canada  48
23        le  quartier  Canada  47
24        de    marche  Canada  44
25   grocery    stores  Canada  44
```

Appendix 6 Bigram counts for Australia

```
> bigram_counts_Australia
       word1        word2   Country   n
1      bondi        beach Australia 116
2     minute         walk Australia 116
3      train      station Australia 110
4    minutes         walk Australia 109
5    walking     distance Australia  88
6    bedroom    apartment Australia  82
7      queen          bed Australia  80
8     sydney          cbd Australia  80
9        min         walk Australia  67
10      size          bed Australia  67
11    public    transport Australia  66
12     queen         size Australia  64
13       bus         stop Australia  60
14  equipped      kitchen Australia  54
15   washing      machine Australia  54
16   central      station Australia  53
17     surry        hills Australia  50
18       air conditioning Australia  49
19     opera        house Australia  48
20   darling      harbour Australia  46
21         2      bedroom Australia  45
22     short         walk Australia  43
23      plan       living Australia  42
24    street      parking Australia  42
25        10      minutes Australia  40
```

Appendix 7 Bigram counts for Brazil

```
> bigram_counts_Brazil
         word1        word2 Country   n
1          rio           de  Brazil 189
2           de      janeiro  Brazil 186
3           pr         ximo  Brazil 161
4           da        praia  Brazil 133
5           wi           fi  Brazil 122
6           ar condicionado  Brazil 110
7          air conditioning  Brazil  98
8           de    copacabana  Brazil  87
9          meu         espa  Brazil  86
10      condom          nio  Brazil  85
11          de        casal  Brazil  85
12       cable           tv  Brazil  81
13          pr          dio  Brazil  79
14          da       tijuca  Brazil  78
15       barra           da  Brazil  74
16        cama           de  Brazil  69
17       todos           os  Brazil  69
18     confort          vel  Brazil  66
19        ximo           ao  Brazil  64
20          da       cidade  Brazil  54
21  copacabana        beach  Brazil  53
22      double          bed  Brazil  53
23       praia           de  Brazil  53
24         rea           de  Brazil  49
25      dispon          vel  Brazil  48
```

Appendix 8 Bigram counts for Portugal

```
> bigram_counts_Portugal
      word1     word2  Country   n
1        da    cidade Portugal 104
2     metro   station Portugal  94
3   walking  distance Portugal  91
4        wi        fi Portugal  77
5      city    center Portugal  74
6        de     banho Portugal  69
7    double       bed Portugal  68
8     santa  catarina Portugal  65
9  equipped   kitchen Portugal  59
10     casa        da Portugal  56
11    douro     river Portugal  54
12       de     metro Portugal  51
13  minutes   walking Portugal  51
14     sofa       bed Portugal  50
15     casa        de Portugal  49
16     free      wifi Portugal  45
17    train   station Portugal  45
18       de     casal Portugal  42
19        5   minutes Portugal  41
20     cama        de Portugal  41
21  minutes      walk Portugal  41
22       de      gaia Portugal  38
23  private  bathroom Portugal  38
24    cable        tv Portugal  36
25  confort       vel Portugal  36
```

Appendix 9 Bigram counts for Hong Kong

```
> bigram_counts_HK
      word1        word2   Country   n
1      hong         kong Hong Kong 377
2       mtr      station Hong Kong 171
3    double          bed Hong Kong 104
4  causeway          bay Hong Kong  96
5      tsim          sha Hong Kong  96
6       sha         tsui Hong Kong  95
7   minutes         walk Hong Kong  91
8      mins         walk Hong Kong  85
9   walking     distance Hong Kong  81
10   sheung          wan Hong Kong  74
11   minute         walk Hong Kong  68
12      min         walk Hong Kong  64
13     free         wifi Hong Kong  63
14  washing      machine Hong Kong  61
15    newly    renovated Hong Kong  55
16      wan         chai Hong Kong  55
17   single          bed Hong Kong  53
18  private     bathroom Hong Kong  50
19     mong          kok Hong Kong  48
20  airport          bus Hong Kong  46
21      sai         ying Hong Kong  45
22     ying          pun Hong Kong  45
23        5         mins Hong Kong  43
24      air conditioning Hong Kong  43
25       wi           fi Hong Kong  43
```

Appendix 10 Bigram counts for Spain

```
> bigram_counts_Spain
      word1        word2 Country   n
1        de           la   Spain 188
2   sagrada      familia   Spain 173
3    double          bed   Spain 131
4  equipped      kitchen   Spain  95
5        en           el   Spain  91
6        de       gracia   Spain  90
7        de    barcelona   Spain  89
8       air conditioning   Spain  78
9       las      ramblas   Spain  75
10     sofa          bed   Spain  75
11    metro      station   Spain  74
12        5      minutes   Spain  73
13  washing      machine   Spain  68
14  minutes      walking   Spain  63
15       la       ciudad   Spain  62
16   single         beds   Spain  60
17  minutes         walk   Spain  59
18       10      minutes   Spain  58
19     city       center   Spain  55
20       wi           fi   Spain  55
21    paseo           de   Spain  54
22    plaza    catalunya   Spain  50
23  walking     distance   Spain  46
24  passeig           de   Spain  44
25       en           la   Spain  43
```

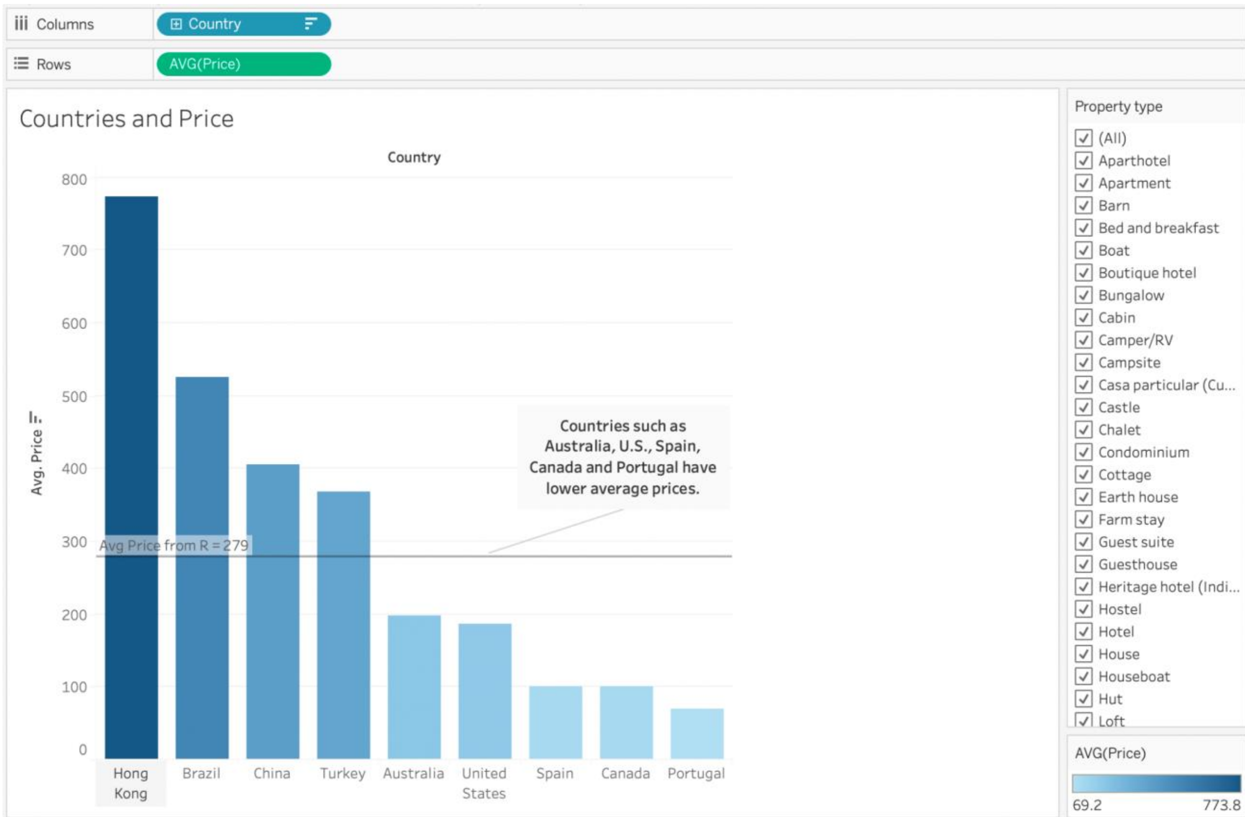
Appendix 11 Bigram counts for China

```
> bigram_counts_China
   word1 word2 Country n
1   7ad9    e2   China 7
2   79bb     3   China 4
3   94c1     2   China 4
4   4f4f     2   China 3
5   884c    10   China 3
6   884c   150   China 3
7   884c     5   China 3
8   94c1     1   China 3
9   94c1     9   China 3
10    12    15   China 2
11  4e50 happy   China 2
12     5     8   China 2
13  533a    50   China 2
14  53e3     9   China 2
15  6709    24   China 2
16  7684    12   China 2
17  79bb     1   China 2
18  79bb     2   China 2
19  79bb    35   China 2
20  7ad9     1   China 2
21  7ad9    10   China 2
22  7ad9    14   China 2
23  7ad9 phone   China 2
24  7ea6   500   China 2
25  884c     3   China 2
```

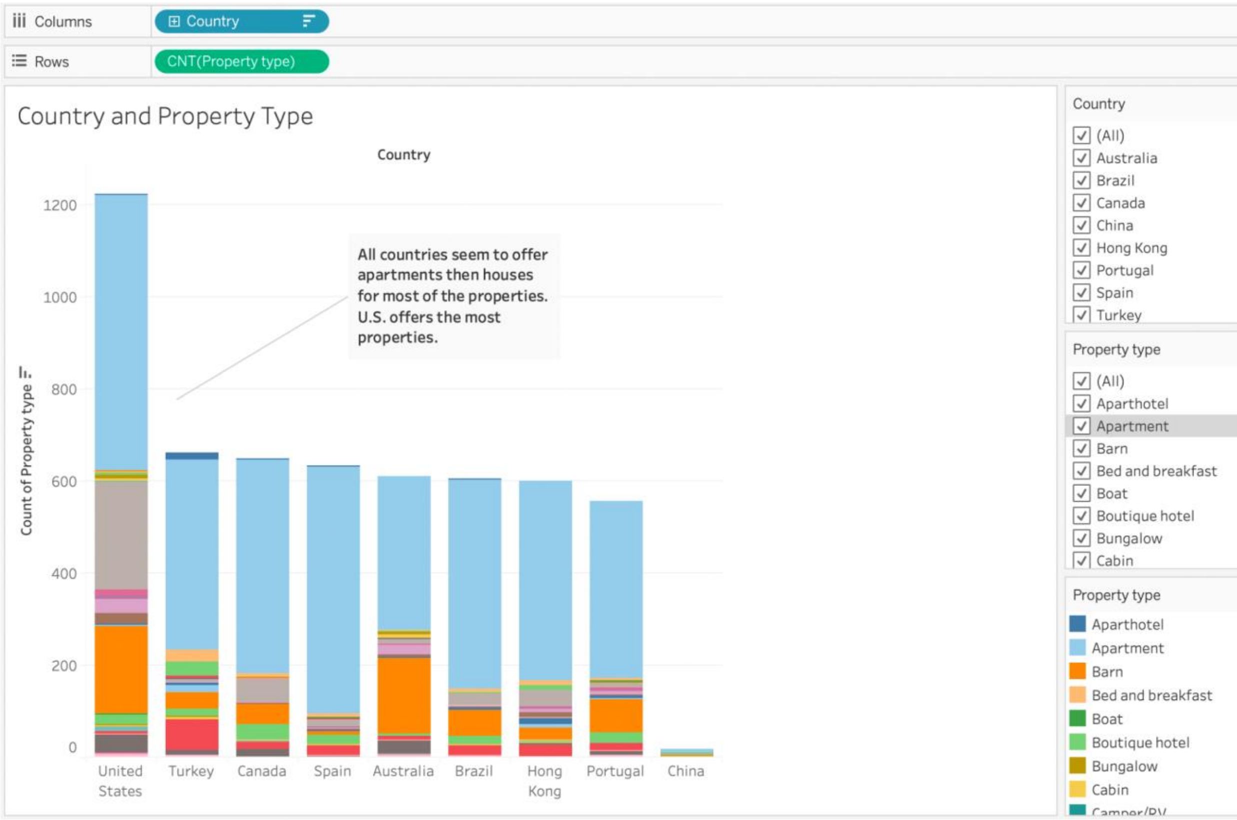
Appendix 12 Bigram counts for Turkey

```
> bigram_counts_Turkey
      word1        word2 Country   n
1   walking     distance  Turkey 122
2    taksim       square  Turkey 105
3         5      minutes  Turkey  87
4  istiklal       street  Turkey  73
5        10      minutes  Turkey  66
6     metro      station  Turkey  64
7        wi           fi  Turkey  54
8      blue       mosque  Turkey  51
9   washing      machine  Turkey  49
10   double          bed  Turkey  48
11   minute         walk  Turkey  45
12  minutes      walking  Turkey  45
13        5          min  Turkey  42
14     city       center  Turkey  42
15   galata        tower  Turkey  42
16  minutes         walk  Turkey  42
17       10          min  Turkey  39
18     <NA>         <NA>  Turkey  37
19    hagia       sophia  Turkey  36
20  topkapi       palace  Turkey  36
21      air conditioning  Turkey  35
22        2      minutes  Turkey  33
23       24        hours  Turkey  30
24        3      minutes  Turkey  28
25     free         wifi  Turkey  28
```

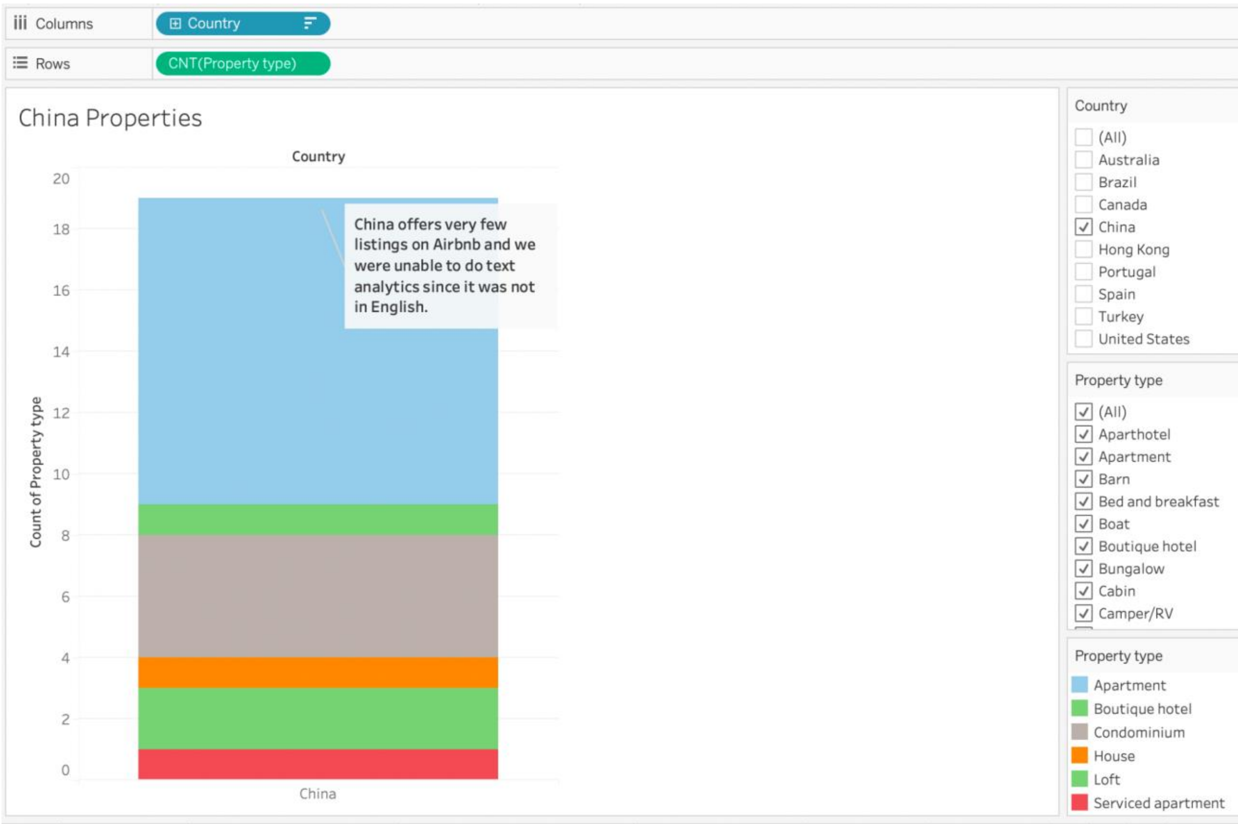
Appendix 13 Average price of listings per country



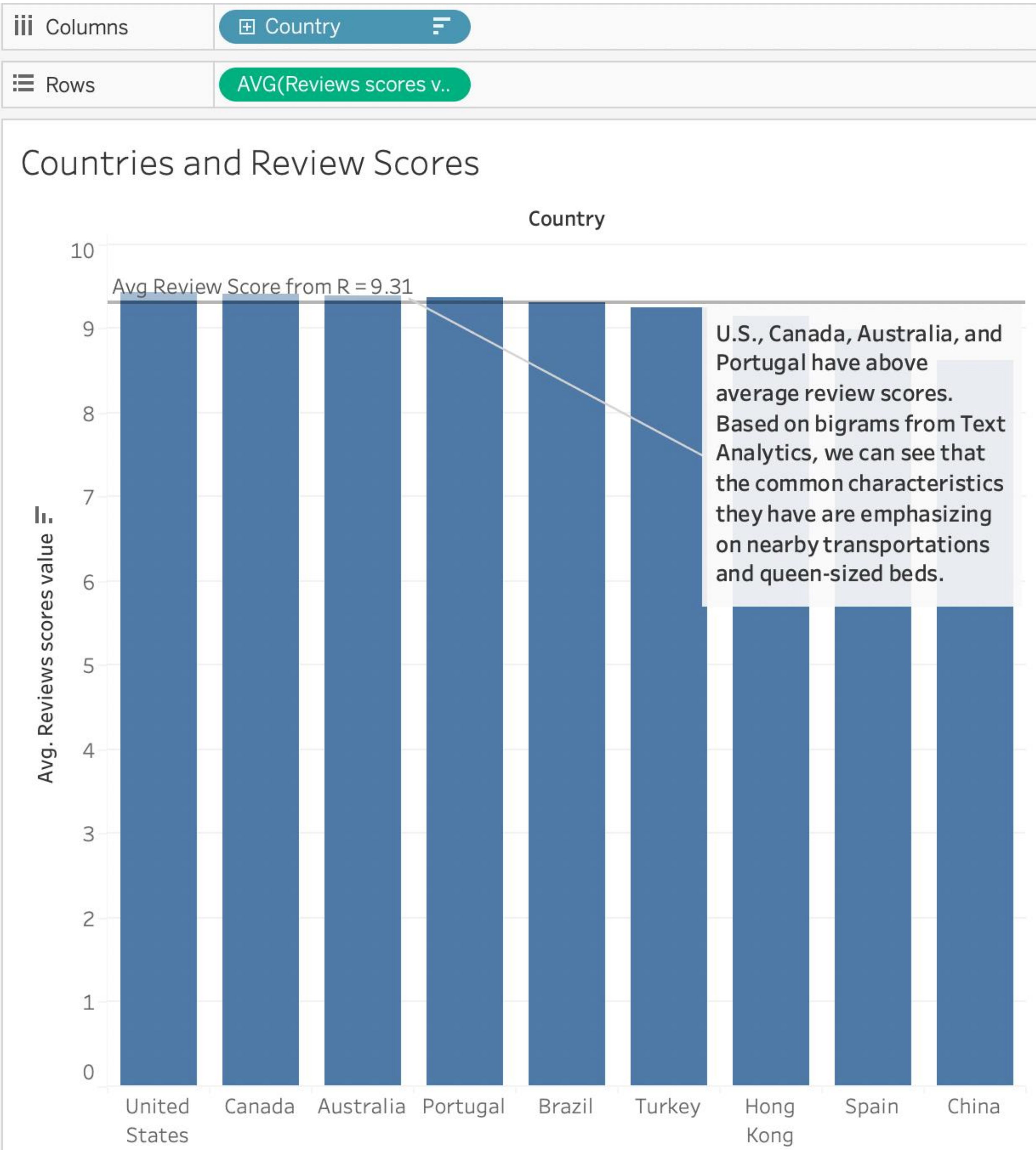
Appendix 14 Counts of each property type by country



Appendix 15 Property types listed in China

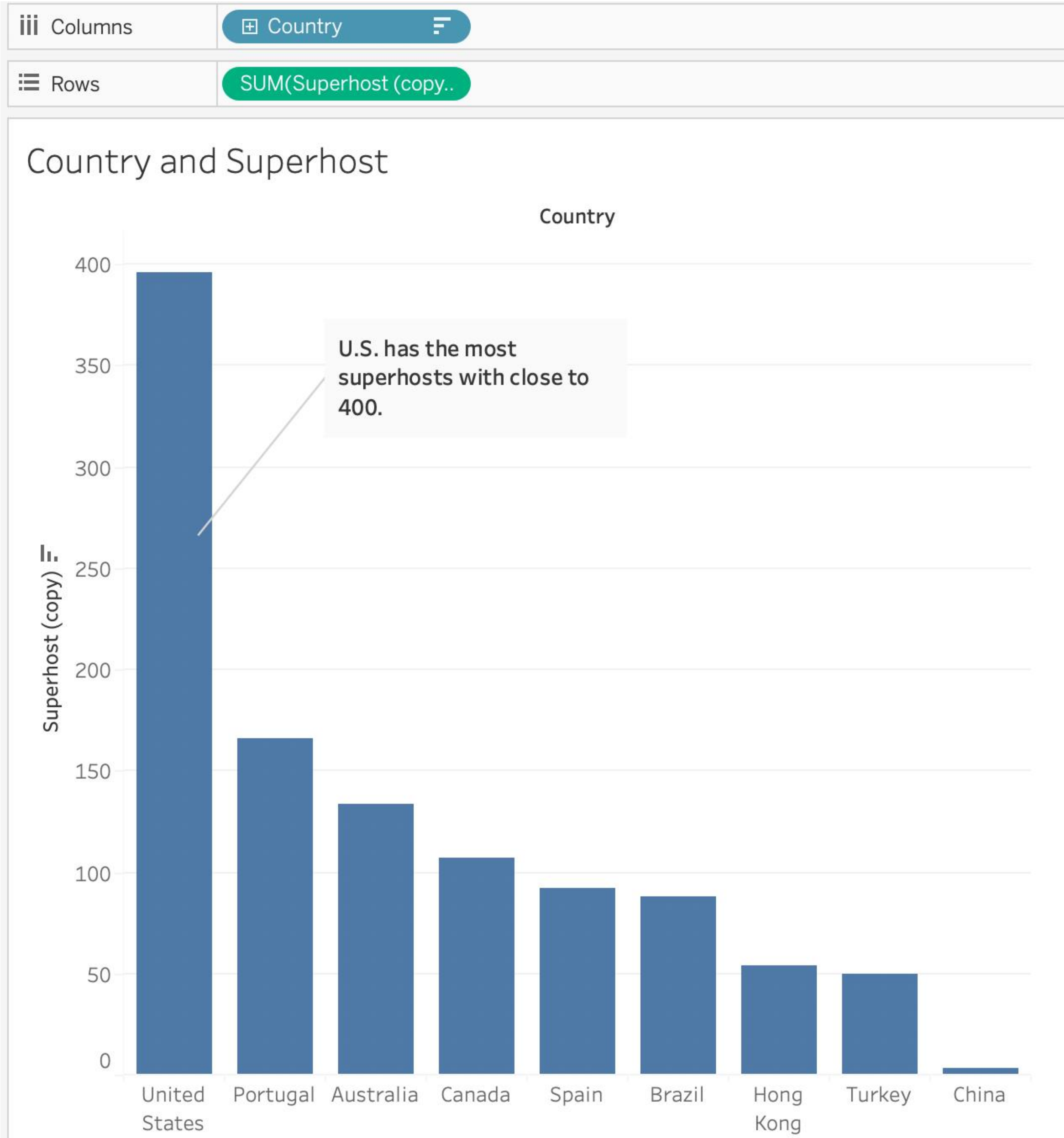


Appendix 16 Average review score per country

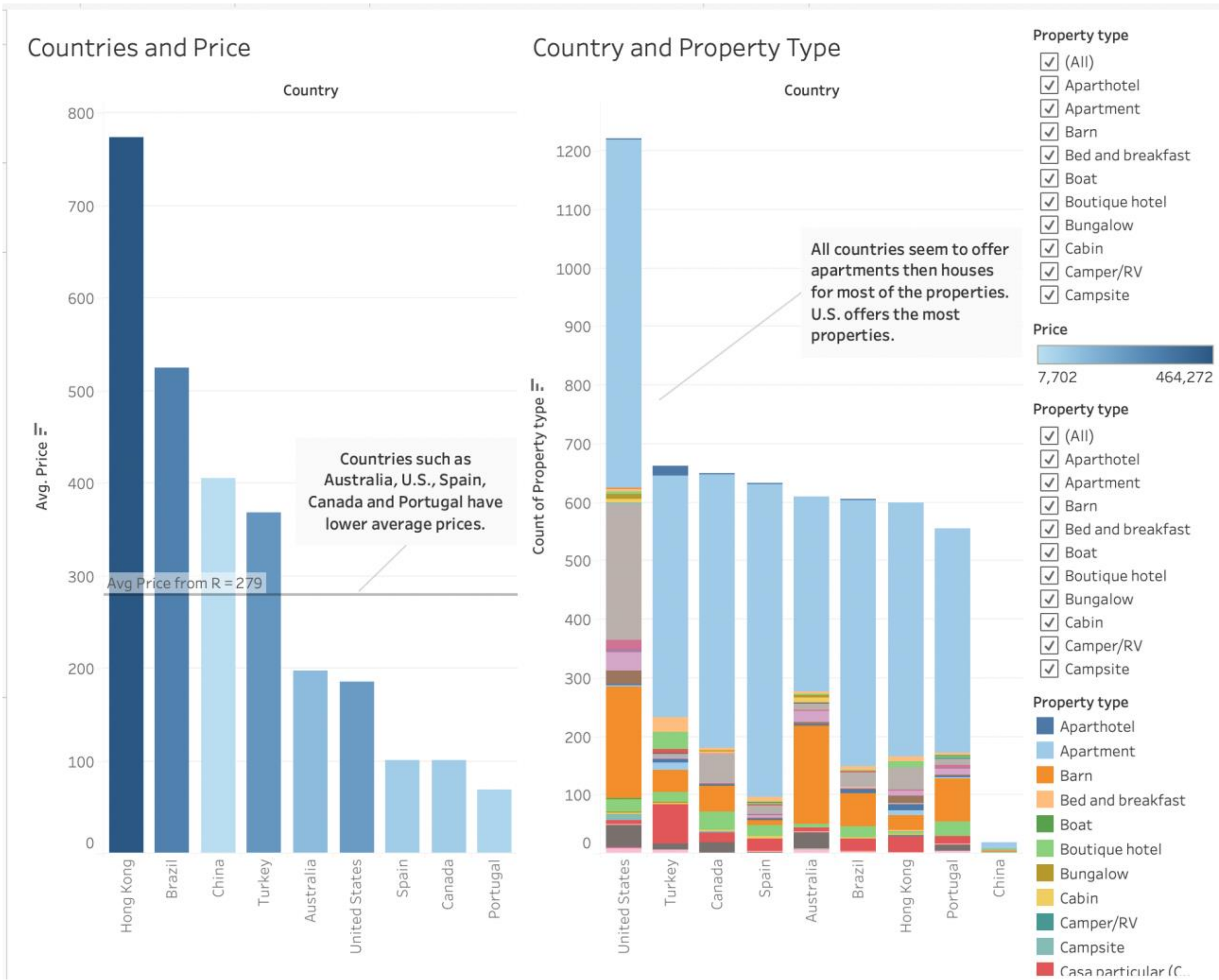


Appendix 17 Number of bedrooms and bathrooms per country

Appendix 18 Number of superhosts per country

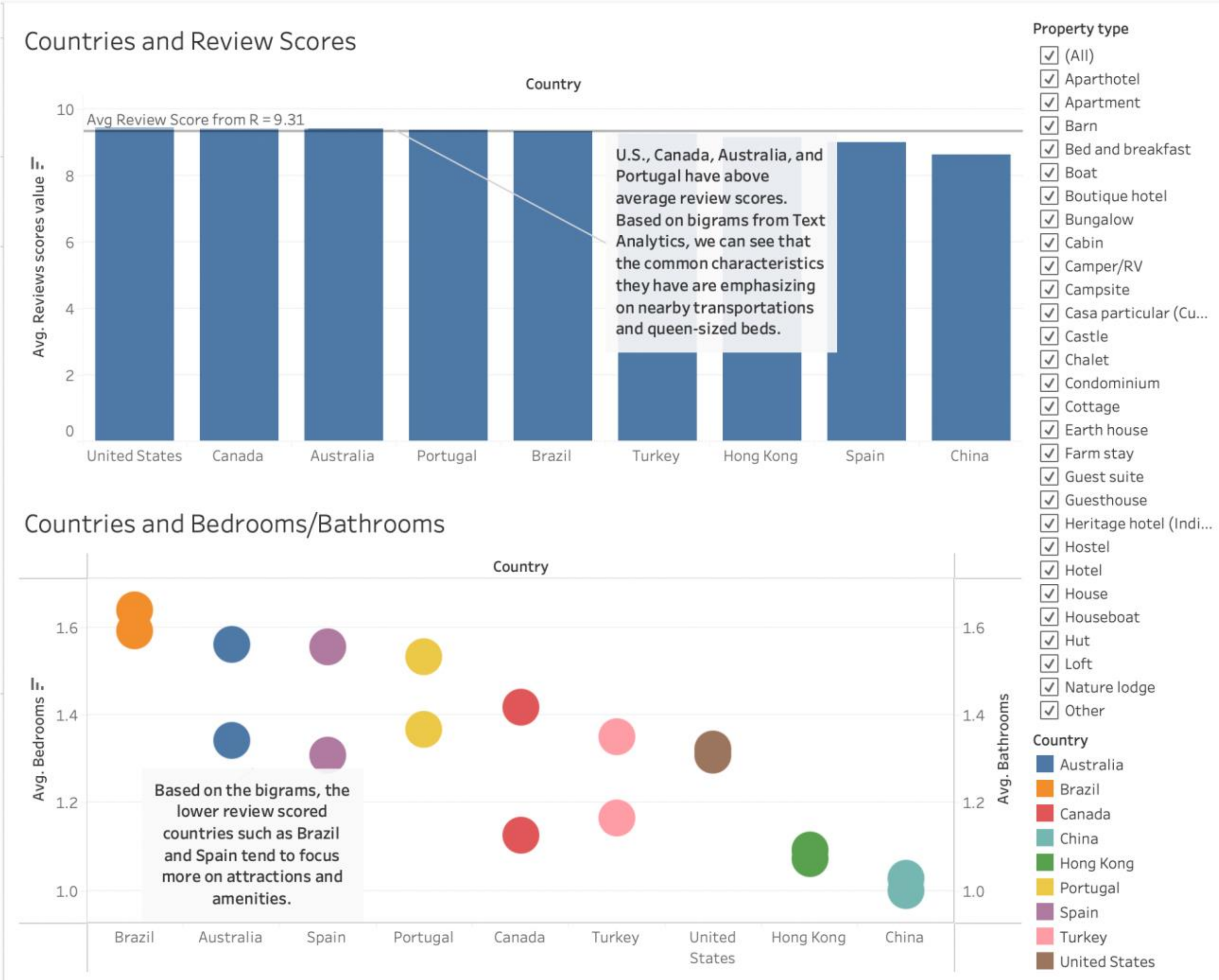


Appendix 19 Dashboard #1



Appendix 20 Dashboard #2

Tableau - Dashboard 2



Appendix 21 R code

```
######################################################
###### Created by Team 5
###### MBAN2 HULT2021
###### Subject: Text Mining
###### Version 0.3
###################################################

#installing and loading the mongolite library to download the Airbnb data
#install.packages("mongolite") #need to run this line of code only once and then you can
comment out
library(mongolite)
library(jsonlite)

# This is the connection_string. You can get the exact url from your MongoDB cluster screen
#replace the <<user>> with your Mongo user name and <<password>> with the mongo
password
#lastly, replace the <<server_name>> with your MongoDB server name
connection_string <-
'mongodb+srv://dpolar96:918o412o@cluster0.prsct.mongodb.net/sample_airbnb?retryWrites=tr
ue&w=majority'
airbnb_collection <- mongo(collection="listingsAndReviews", db="sample_airbnb",
url=connection_string)

#Here's how you can download all the Airbnb data from Mongo
## keep in mind that this is huge and you need a ton of RAM memory

airbnb_all <- airbnb_collection$find()
#######################################################
######################################################
#1 subsetting your data based on a condition:
Name <- airbnb_all$name
Descritpion <- airbnb_all$description
Property_type <- airbnb_all$property_type
Room_type <- airbnb_all$room_type
Guest.number <- airbnb_all$accommodates
Bedrooms <- airbnb_all$bedrooms
Beds <- airbnb_all$beds
Bathrooms <- airbnb_all$bathrooms
Reviews_scores_value <- airbnb_all$review_scores$review_scores_value
airbnb_verified <- airbnb_all$host$host_verifications
Superhost <- airbnb_all$host$host_is_superhost
Host <- airbnb_all$host$host_name
```

```
ID <- airbnb_all$host$host_id
Country <- airbnb_all$address$country
City <- airbnb_all$address$market
Cancellation <- airbnb_all$cancellation_policy
Price <- airbnb_all$price
Weekly_price <- airbnb_all$weekly_price
Monthly_price <- airbnb_all$monthly_price

airbnb_work <- cbind(Name, Descritpion, Property_type, Room_type,
                     Guest.number, Bedrooms, Beds, Bathrooms, Reviews_scores_value,
                     Superhost, Host, Country, City, Cancellation, ID, Price,
                     Weekly_price, Monthly_price)

write.csv(airbnb_work, "C:/Users/polar/Downloads/airbnb.csv")

#Downloading necessary packages
library(tidytext)
library(tidyverse)
library(tidyr)
library(tidytuesdayR)
library(stringr)
library(textreadr)
library(pdftools)
library(textshape)
library(twitteR)
library(tm)
library(ggplot2)
library(scales)
library(magrittr)
library(dplyr)
library(gutenbergr)
library(Matrix)
library(textdata)
library(igraph)
library(ggraph)
library(widyr)
library(topicmodels)
library(gutenbergr)
library(quanteda)
library(quanteda.textmodels)
library(RColorBrewer)
library(tibble)
library(stringr)
```

```
airbnb <- read_csv("/Users/tsztinviviansoo/Desktop/combined project/airbnb.csv")
View(airbnb)

data <- c(airbnb[1],airbnb[3],airbnb[13])
data<- data.frame(data)
data
airbnb<-data.frame(airbnb)
colnames(airbnb)[1] <- "IDme"
colnames(airbnb)[3] <- "text"

#########################################

airbnb_token <- airbnb %>%
  unnest_tokens(word, text)

nrcpositive <- get_sentiments("nrc") %>%
  filter(sentiment == "positive")

mean(airbnb$Price)
summary(airbnb$Price)

#mean price is $279 but Q3 is $280 too which means the data is skewed right- median price
$129
#let us see if a higher price has more positive sentiment


high<-airbnb_token %>%
  filter(Price >= 279) %>%        #taking $279 as a reference point allows us to look at the highest
sentiment words for the top 25% listing prices
  inner_join(nrcpositive) %>%
  count(word, sort=T)
high

low<- airbnb_token %>%
  filter(Price <= 129) %>%        #taking $129 as a reference point allows us to look at the highest
sentiment words for the bottom 25% listing prices
  inner_join(nrcpositive) %>%
  count(word, sort=T)
low

#R <=129                >=279
#1 building    686      #1 Building
#2 quiet       678      #2 enjoy
#3 fully       659      #3 shopping
```

```
#4 center      582      #4 fully
#5 full        567      #5 quiet
#6 beautiful   566      #6 spacious
#7 enjoy       564      #7 full
#8 spacious    431      #8 pool
#9 perfect     405      #9 beautiful
#10 clean      379      #10 perfect

#we noticed that for the higher price range, there is an added need for top sentiment, which are shopping and pool
################################################

###############Sentiment Analysis###################

### For the sentiment analysis we will be looking at English speaking countries

#Filtering by country - Australia
Australia <- airbnb_token %>%
  filter(Country == "Australia") %>%
  anti_join(stop_words) %>%
  count(word, sort=T)

Australia

# Top 10 words and frequency
#            word   n
#1      apartment 693
#2           walk 530
#3          beach 476
#4        bedroom 472
#5         sydney 459 #Listings in Sydney are more than any other city
#6        kitchen 453
#7          house 391
#8            bed 375
#9           city 353
#10             2 347

#Filtering by country - Australia and prices above average ($279)
Australia_high <- airbnb_token %>%
  filter(Country == "Australia" & Price >= 279) %>%
  anti_join(stop_words) %>%
  count(word, sort=T)

Australia_high
```

```
afinn_Australia_high <- Australia_high %>%
  inner_join(get_sentiments("afinn"))%>%
  summarise(sentiment=sum(value)) %>%
  mutate(method="AFINN")

bing_and_nrc_Australia_high <- bind_rows(
  Australia_high%>%
    inner_join(get_sentiments("bing"))%>%
    mutate(method = "Bing et al."),
  Australia_high %>%
    inner_join(get_sentiments("nrc") %>%
                 filter(sentiment %in% c("positive", "negative"))) %>%
    mutate(method = "NRC")) %>%
  count(method,  sentiment) %>%
  spread(sentiment, n, fill=0) %>%
  mutate(sentiment = positive-negative)

bind_rows(afinn_Australia_high, bing_and_nrc_Australia_high)

## AFINN = 198
## Bing = 145 (Positive = 186 and Negative = 41)
## NRC = 164 (Positive = 234 and Negative = 70)

# Overall more positive than negative sentiments for Airbnbs above the average price point

#Top 10 words for Australia_high
#word   n
#1          beach 142 #people are willing to pay more if the Airbnb is closer to the beach
#2           walk 111
#3           home  99
#4        bedroom  98
#5          house  95
#6              2  82
#7         living  82
#8         sydney  82
#9      apartment  81
#10       kitchen  79

#######

#Filtering by country - Australia and prices below $129
Australia_low <- airbnb_token %>%
  filter(Country == "Australia" & Price <= 129) %>%
```

```
  anti_join(stop_words) %>%
  count(word, sort=T)

Australia_low

afinn_Australia_low <- Australia_low %>%
  inner_join(get_sentiments("afinn"))%>%
  summarise(sentiment=sum(value)) %>%
  mutate(method="AFINN")

bing_and_nrc_Australia_low <- bind_rows(
  Australia_low%>%
    inner_join(get_sentiments("bing"))%>%
    mutate(method = "Bing et al."),
  Australia_low %>%
    inner_join(get_sentiments("nrc") %>%
            filter(sentiment %in% c("positive", "negative"))) %>%
    mutate(method = "NRC")) %>%
  count(method,  sentiment) %>%
  spread(sentiment, n, fill=0) %>%
  mutate(sentiment = positive-negative)

bind_rows(afinn_Australia_low, bing_and_nrc_Australia_low)

## AFINN = 265
## Bing = 170 (Positive = 245 and Negative = 75)
## NRC = 239 (Positive = 343 and Negative = 104)

# Overall more positive than negative sentiments for Airbnbs below the average price point
## More positive sentiments in the lower price range than high, suggesting that more people book cheaper Airbnbs
### Low cost apartments by the beach seem to be enticing for customers in Australia

#Top 10 words for Australia_low
#          word   n
#1     apartment 281
#2          walk 250
#3        sydney 222
#4       kitchen 219 #Basic amenities are reviewed more in cheaper airbnbs to see if needs aren't compensated for
#5         house 209
#6       bedroom 189
#7          city 181
#8         beach 180
```

```
#9          bed 179
#10    bathroom 168

#######

#Filtering by country - Canada
Canada <- airbnb_token %>%
  filter(Country == "Canada") %>%
  anti_join(stop_words) %>%
  count(word, sort=T)
Canada

#Looking at Top 10 frequent words after ignoring French stop words - de, la, le, du, est, vous
and des
# Top 10 words and frequency
#          word   n
#1    apartment 471
#2     montreal 400 #Listings in Montreal more than any other city
#3            2 377
#4      minutes 351
#5  restaurants 333
#6      located 310
#7        metro 299
#8      kitchen 294
#9         walk 267
#10         bed 257

#Filtering by country - Canada and prices above average ($279)
Canada_high <- airbnb_token %>%
  filter(Country == "Canada" & Price >= 279)%>%
  anti_join(stop_words) %>%
  count(word, sort=T)

Canada_high

afinn_Canada_high <- Canada_high %>%
  inner_join(get_sentiments("afinn"))%>%
  summarise(sentiment=sum(value)) %>%
  mutate(method="AFINN")

bing_and_nrc_Canada_high <- bind_rows(
  Canada_high%>%
    inner_join(get_sentiments("bing"))%>%
    mutate(method = "Bing et al."),
```

```
  Canada_high %>%
    inner_join(get_sentiments("nrc") %>%
               filter(sentiment %in% c("positive", "negative"))) %>%
    mutate(method = "NRC")) %>%
  count(method,  sentiment) %>%
  spread(sentiment, n, fill=0) %>%
  mutate(sentiment = positive-negative)

bind_rows(afinn_Canada_high, bing_and_nrc_Canada_high)

## AFINN = 67
## Bing = 54 (Positive = 65 and Negative = 11)
## NRC = 69 (Positive = 80 and Negative = 11)

# Overall more positive than negative sentiments for Airbnbs above the average price point

#Top 10 words for Canada_high

#word  n
#1       montreal 20
#2       downtown 18
#3              3 16
#4            bed 16
#5        bedroom 16
#6      apartment 15
#7              2 14
#8          house 12
#9        located 12
#10         floor 11

#######

#Filtering by country - Canada and prices prices below $129
Canada_low <- airbnb_token %>%
  filter(Country == "Canada" & Price <= 129)%>%
  anti_join(stop_words) %>%
  count(word, sort=T)

Canada_low

afinn_Canada_low <- Canada_low  %>%
  inner_join(get_sentiments("afinn"))%>%
  summarise(sentiment=sum(value)) %>%
  mutate(method="AFINN")
```

```
bing_and_nrc_Canada_low <- bind_rows(
  Canada_low %>%
    inner_join(get_sentiments("bing"))%>%
    mutate(method = "Bing et al."),
  Canada_low  %>%
    inner_join(get_sentiments("nrc") %>%
                 filter(sentiment %in% c("positive", "negative"))) %>%
    mutate(method = "NRC")) %>%
  count(method,  sentiment) %>%
  spread(sentiment, n, fill=0) %>%
  mutate(sentiment = positive-negative)

bind_rows(afinn_Canada_low, bing_and_nrc_Canada_low)

## AFINN = 261
## Bing = 193 (Positive = 267 and Negative = 74)
## NRC = 266 (Positive = 375 and Negative = 109)

# Overall more positive than negative sentiments for Airbnbs below the average price point
## More positive sentiments in lower price range than high, suggesting that more people booked cheaper Airbnbs
### Canada had the lowest score for Airbnbs at the high price range
#### Low cost apartments in Montreal seem enticing to customers

#Top 10 words for Canada_low
#            word   n
#1      apartment 357
#2        minutes 293
#3       montreal 286
#4              2 280
#5    restaurants 268
#6          metro 258
#7        located 236
#8        kitchen 231
#9           walk 220
#10       station 207

#######

#Filtering by country - United States
United_States <- airbnb_token %>%
  filter(Country == "United States") %>%
  anti_join(stop_words) %>%
```

```
  count(word, sort=T)
United_States

# Top 10 words and frequency
#          word    n
#1       bedroom 1112
#2     apartment 1023
#3       kitchen  978
#4         beach  838
#5           bed  818
#6             2  778
#7       private  764
#8        living  743
#9       located  690
#10         home  617

#Filtering by country - United States and prices above average ($279)
United_States_high <- airbnb_token %>%
  filter(Country == "United States" & Price >= 279)%>%
  anti_join(stop_words) %>%
  count(word, sort=T)

United_States_high

United_States_afinn_high <- United_States_high %>%
  inner_join(get_sentiments("afinn"))%>%
  summarise(sentiment=sum(value)) %>%
  mutate(method="AFINN")

United_States_bing_and_nrc_high <- bind_rows(
  United_States_high%>%
    inner_join(get_sentiments("bing"))%>%
    mutate(method = "Bing et al."),
  United_States_high %>%
    inner_join(get_sentiments("nrc") %>%
              filter(sentiment %in% c("positive", "negative"))) %>%
    mutate(method = "NRC")) %>%
  count(method,  sentiment) %>%
  spread(sentiment, n, fill=0) %>%
  mutate(sentiment = positive-negative)

bind_rows(United_States_afinn_high, United_States_bing_and_nrc_high)

## AFINN = 305
```

```
## Bing = 200 (Positive = 260 and Negative = 60)
## NRC = 289 (Positive = 382 and Negative = 93)

# Overall more positive than negative sentiments for Airbnbs above the average price point

#Top 10 words for United_States_high
#         word   n
#1       beach 211
#2     bedroom 207
#3       ocean 176
#4           2 172
#5      living 145
#6        home 139
#7     kitchen 138
#8     located 127
#9       views 115
#10          3 103

#######

#Filtering by country - United States and prices below $129
United_States_low <- airbnb_token %>%
  filter(Country == "United States" & Price <= 129)%>%
  anti_join(stop_words) %>%
  count(word, sort=T)

United_States_low

United_States_afinn_low <- United_States_low %>%
  inner_join(get_sentiments("afinn"))%>%
  summarise(sentiment=sum(value)) %>%
  mutate(method="AFINN")

United_States_bing_and_nrc_low <- bind_rows(
  United_States_low%>%
    inner_join(get_sentiments("bing"))%>%
    mutate(method = "Bing et al."),
  United_States_low %>%
    inner_join(get_sentiments("nrc") %>%
                 filter(sentiment %in% c("positive", "negative"))) %>%
    mutate(method = "NRC")) %>%
  count(method,  sentiment) %>%
  spread(sentiment, n, fill=0) %>%
  mutate(sentiment = positive-negative)
```

```
bind_rows(United_States_afinn_low, United_States_bing_and_nrc_low )

## AFINN = 346
## Bing = 214 (Positive = 358 and Negative = 144)
## NRC = 343 (Positive = 522 and Negative = 179)

# Overall more positive than negative sentiments for Airbnbs below the average price point
## More positive sentiments in lower price range than high, suggesting that more people
booked cheaper Airbnbs
### Many seem to opt for low cost bedrooms than entire apartments in the United States

#Top 10 words for United_States_low
#word   n
#1      apartment 566
#2        bedroom 469
#3        kitchen 463
#4        private 447
#5            bed 429
#6       bathroom 355
#7              2 342
#8         living 339
#9           walk 339
#10       located 327

#######

###############BIGRAM##################
#Creating bigram of comments
airbnb_bigrams <- data %>%
  unnest_tokens(bigram, Descritpion, token = "ngrams", n=2)

airbnb_bigrams

airbnb_bigrams %>%
  count(bigram, sort = TRUE) #this has many stop words, need to remove them

#to remove stop words we need to separate each word then remove:

bigrams_separated <- airbnb_bigrams %>%
  separate(bigram, c("word1", "word2"), sep = " ")

bigrams_filtered <- bigrams_separated %>%
  filter(!word1 %in% stop_words$word) %>%
```

```
  filter(!word2 %in% stop_words$word)

#creating the new bigram, "no-stop-words":
bigram_counts <- bigrams_filtered %>%
  count(word1, word2, Country, sort = TRUE)
bigram_counts%>%
  head(25)

bigram_counts <- bigrams_filtered %>%
  count(word1, word2, Country, sort = TRUE)%>%
  head(25)
bigram_counts%>%
  head(25)

bigram_counts_US <- bigrams_filtered %>%
  count(word1, word2, Country, sort = TRUE)%>%
  filter(Country=="United States") %>%
head(25)

bigram_counts_Canada <- bigrams_filtered %>%
  count(word1, word2, Country, sort = TRUE)%>%
  filter(Country=="Canada") %>%
  head(25)

bigram_counts_Australia <- bigrams_filtered %>%
  count(word1, word2, Country, sort = TRUE)%>%
  filter(Country=="Australia") %>%
  head(25)

bigram_counts_Brazil <- bigrams_filtered %>%
  count(word1, word2, Country, sort = TRUE)%>%
  filter(Country=="Brazil") %>%
  head(25)

bigram_counts_Portugal <- bigrams_filtered %>%
  count(word1, word2, Country, sort = TRUE)%>%
  filter(Country=="Portugal") %>%
  head(25)

bigram_counts_HK <- bigrams_filtered %>%
  count(word1, word2, Country, sort = TRUE)%>%
  filter(Country=="Hong Kong") %>%
  head(25)
```

```
bigram_counts_Spain <- bigrams_filtered %>%
  count(word1, word2, Country, sort = TRUE)%>%
  filter(Country=="Spain") %>%
  head(25)

bigram_counts_China <- bigrams_filtered %>%
  count(word1, word2, Country, sort = TRUE)%>%
  filter(Country=="China") %>%
  head(25)

bigram_counts_Turkey <- bigrams_filtered %>%
  count(word1, word2, Country, sort = TRUE)%>%
  filter(Country=="Turkey") %>%
  head(25)
###############TRIGRAM###################

airbnb_trigrams <- data %>%
  unnest_tokens(trigram, Descritpion, token = "ngrams", n=3)

airbnb_trigrams

airbnb_trigrams %>%
  count(trigram, sort = TRUE) #this has many stop words, need to remove them

#to remove stop words we need to separate each word then remove:

trigrams_separated <- airbnb_trigrams %>%
  separate(trigram, c("word1", "word2","word3"), sep = " ")

trigrams_filtered <- trigrams_separated %>%
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word) %>%
  filter(!word3 %in% stop_words$word)

#creating the new trigram, "no-stop-words":
trigram_counts <- trigrams_filtered %>%
  count(word1, word2,word3, Country, sort = TRUE)
trigram_counts%>%
  head(25)

###############QUADROGRAM###################

airbnb_quadrograms <- data %>%
```

```
  unnest_tokens(quadrogram, Descritpion, token = "ngrams", n=4)

airbnb_quadrograms

airbnb_quadrograms %>%
  count(quadrogram, sort = TRUE) #this has many stop words, need to remove them

#to remove stop words we need to separate each word then remove:

quadrograms_separated <- airbnb_quadrograms %>%
  separate(quadrogram, c("word1", "word2","word3","word4"), sep = " ")

quadrograms_filtered <- quadrograms_separated %>%
  filter(!word1 %in% stop_words$word) %>%
  filter(!word2 %in% stop_words$word) %>%
  filter(!word3 %in% stop_words$word) %>%
  filter(!word4 %in% stop_words$word)

#creating the new quadrogram, "no-stop-words":
quadrogram_counts <- quadrograms_filtered %>%
  count(word1, word2,word3,word4, Country, sort = TRUE)
quadrogram_counts%>%
  head(25)

#highest quadrogram count is n=34 / no high business insight / not making more sense




### creating a tidy format for US  apartments
usa <- airbnb %>%
  filter(Country== "United States" & Property_type=="Apartment")

tidy_usa <- usa %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words)
print(tidy_usa)

### creating a tidy format for Spain apartments
spain <- airbnb %>%
  filter(Country== "Spain" & Property_type=="Apartment")
```

```
tidy_spain <- spain %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words)
print(tidy_spain)

### creating a tidy format for Canada apartments
canada <- airbnb %>%
  filter(Country== "Canada" & Property_type=="Apartment")

tidy_canada <- canada %>%
  unnest_tokens(word, text) %>%
  anti_join(stop_words)
print(tidy_canada)

##############################################
####We want to combine all the datasets and do frequencies
##############################################
library(tidyr)
library(stringr)
frequency <- bind_rows(mutate(tidy_usa, author="United States"),
                       mutate(tidy_canada, author= "Canada"),
                       mutate(tidy_spain, author="Spain")
)%>%#closing bind_rows
  mutate(word=str_extract(word, "[a-z']+")) %>%
  count(author, word) %>%
  group_by(author) %>%
  mutate(proportion = n/sum(n))%>%
  select(-n) %>%
  spread(author, proportion) %>%
  gather(author, proportion, `Canada`, `Spain`)

#let's plot the correlograms:
library(scales)
library(ggplot2)
ggplot(frequency, aes(x=proportion, y=`United States`,
                      color = abs(`United States`- proportion)))+
  geom_abline(color="grey40", lty=2)+
  geom_jitter(alpha=.1, size=2.5, width=0.3, height=0.3)+
  geom_text(aes(label=word), check_overlap = TRUE, vjust=1.5) +
  scale_x_log10(labels = percent_format())+
  scale_y_log10(labels= percent_format())+
  scale_color_gradient(limits = c(0,0.001), low = "darkslategray4", high = "gray75")+
  facet_wrap(~author, ncol=2)+
  theme(legend.position = "none")+
```

```
  labs(y= "United States", x=NULL)

###########################################
##doing the cor.test() ################
###########################################

cor.test(data=frequency[frequency$author == "Canada",],
        ~proportion + `United States`)

cor.test(data=frequency[frequency$author == "Spain",],
        ~proportion + `United States`)
```